



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV KONSTRUOVÁNÍ

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF MACHINE AND INDUSTRIAL DESIGN

NÁVRH LETADLOVÉ ENERGETICKÉ JEDNOTKY

DESIGN OF AIRCRAFT POWER UNIT

PŘÍLOHA „B“ K DIPLOMOVÉ PRÁCI

APPENDIX B

AUTOR PRÁCE
AUTHOR

Bc. MARTIN POLEDNO

VEDOUcí PRÁCE
SUPERVISOR

prof. RNDr. Ing. JOSEF NEVRLÝ, CSc.

BRNO 2010

Tato příloha k diplomové práci obsahuje výpis zdrojového kódu všech m-skriptů, které byly pro účely diplomové práce vytvořeny.

OBSAH

TEPELNÝ OBĚH TURBOHŘÍDELOVÉHO MOTORU (kap. 5)	
<i>obeh.m</i>	5
NÁVRH ZÁKLADNÍCH ROZMĚRŮ Odstředivého kompresoru (kap. 6.1)	
<i>kompresor.m</i>	8
CHARAKTERISTIKA Odstředivého kompresoru (kap. 6.2)	
<i>komp_chka.m</i>	15
<i>nactidiagramy.m</i>	18
<i>cti.m</i>	19
<i>ctiDG1.m</i>	19
NÁVRH ZÁKLADNÍCH ROZMĚRŮ OSOVÉ TURBÍNY (kap. 7.1)	
<i>turbina.m</i>	20
<i>rozdeleni.m</i>	26
CHARAKTERISTIKA TURBÍNY (kap. 7.2)	
<i>prepocet_turb.m</i>	27
CHARAKTERISTIKA VÝSTUPNÍ SOUSTAVY (kap. 8.2)	
<i>tryska.m</i>	29
MEZINÁRODNÍ STANDARDNÍ ATMOSFÉRA (kap. 9.1)	
<i>MSA.m</i>	30
VÝŠKOVÁ CHARAKTERISTIKA MOTORU (kap. 9.2)	
<i>vyskova_chka.m</i>	31
STANOVENÍ UŽITEČNÉHO VÝKONU MOTORU V NEVÝP. REŽIMECH (kap. 10)	
<i>vypocetPef.m</i>	33
<i>izocary.m</i>	35
<i>cti_komp.m</i>	36
<i>cti_turb.m</i>	36
<i>cti_trysk.m</i>	37
<i>nacti_komp.m</i>	37
<i>nacti_turb.m</i>	38
<i>nacti_tr.m</i>	38
<i>najdi_rozsah.m</i>	39
<i>vysledky.m</i>	39


```
% VYPOCET OBEHU TURBOHRIDELOVEHO MOTORU
% Bc. Martin Poledno, červen 2008
% -----

clc;clear;

% Zakladni udaje o obehu
% -----
pi_kc = 4.1;           % [-]          stlacení kompresoru
T_3c = 1000;          % [K]          teplota plynu za spalovací komorou
Qm = 1.1;              % [kg/s]       hmotnostní průtok vzduchu motorem

% Fyzikalni vlastnosti
% -----
kappa_v = 1.4;         % [-]          Poissonova konstanta vzduchu
kappa_pl = 1.33;        % [-]          Poissonova konstanta plynu
c_pv = 1005;           % [J/(kg*K)] izobarická mer. tep. kapacita vzduchu
c_ppl = 1158;          % [J/(kg*K)] izobarická mer. tep. kapacita plynu

% Odhadovane a volene hodnoty
% -----
sigma_vst = 0.985;     % [-]          součinitel zachování celk. tlaku ve vstupní soustavě
sigma_sk = 0.96;       % [-]          součinitel zachování celk. tlaku ve spalovací komoře
eta_kc = 0.80;         % [-]          celková účinnost kompresoru
eta_tc = 0.85;         % [-]          celková účinnost turbíny
eta_m = 0.98;          % [-]          mechanická účinnost rotoru
lambda = 1.02;         % [-]          součinitel, odpovídající změně c_ppl mezi stavy 5 a 6
fi_tr = 0.98;          % [-]          součinitel ztráty rychlosti ve výstupní trysce
c_6 = 65;              % [m/s]       rychlost výtoku plynu z výstupní trysky

% Pro nadmorskou výšku H=0 platí:
p_0 = 101325;          % [Pa]          standardní atmosferický tlak
T_0 = 288;             % [K]          standardní atmosferická teplota

M_0 = 0;               % [-]          rychlost letu (Machovo číslo)

disp('=====');
disp('                VYPOCET TEPELNEHO OBEHU MOTORU                ');
disp('=====');

% (0)
% Celkový tlak a celková teplota vzduchu před motorem
p_0c = p_0 * (1 + (kappa_v-1)/2 * (M_0)^2)^(kappa_v/(kappa_v-1));
T_0c = T_0 * (1 + (kappa_v-1)/2 * (M_0)^2);

% (1)
% Parametry vzduchu na vstupu do kompresoru:
p_1c = p_0c * sigma_vst;
T_1c = T_0c;

% (2)
% Parametry vzduchu na výstupu z kompresoru:
p_2c = p_1c * pi_kc;
T_2c = T_1c * (1 + (pi_kc^((kappa_v-1)/kappa_v) - 1) / eta_kc);
```

```

% (3)
% Parametry vzduchu na vystupu ze spalovaci komory:
p_3c = p_2c * sigma_sk;
% T_3c zadano

% (4)
% Expanze plyna na turbine
T_4c = T_3c - c_pv/(c_pp1 * eta_m) * (T_2c - T_1c) * 1; % rovnovaha praci na hrideli
p_4c = p_3c * (1 - (1 - T_4c/T_3c)/ eta_tc)^(kappa_pl/(kappa_pl-1));

% Adiabaticka teplota plynu na vystupu z motoru
T_6ad = T_4c * ( p_0 / p_4c )^((kappa_pl-1)/kappa_pl);

% Adiabaticka teplota plynu na vystupu z turbiny
T_5cad = T_6ad + c_6^2 / (2*c_pp1*fi_tr^2*lambda);

% (5)
% Celkove parametry plynu za turbinou
T_5c = T_4c - (T_4c - T_5cad)*eta_tc;
p_5c = p_4c * (T_5cad/T_4c)^(kappa_pl/(kappa_pl-1));

% Uzitecna merna prace na hrideli motoru
W_ef = c_pp1 * (T_4c - T_5c);
disp('Efektivni merna prace na hrideli motoru W_ef [J/kg]');
disp(W_ef);

% Uzitecny vykon na hrideli motoru
P_ef = W_ef * Qm;
disp('Efektivni vykon na hrideli motoru P_ef [W]');
disp(P_ef);

% (6)
% Expanze na vystupni trysce
p_6c = p_0c;
T_6c = T_5c;

% at_turb = c_pp1 * (T_4c - T_3c) % merna technicka prace turbiny
% at_komp = c_pv * (T_2c - T_1c) % merna technicka prace kompresoru

% VYKRESLENI p-v DIAGRAMU OBEHU TURBOHRIDELOVEHO MOTORU
% -----

% Vytvoreni a popis grafu
% -----
rozsah_x = 2.3;
rozsah_y = 450000;
figure('Color',[1 1 1]);
axes('XGrid','on','YGrid','on','XTick',[0:0.25:rozsah_x],'XMinorTick','off',...
    'YTick',[50000:50000:rozsah_y],'YMinorTick','off','LineWidth',2,...
    'FontSize',14);
xlim([0,rozsah_x]);
xlabel('\fontsize{14}v [m^3.kg^{-1}]');
ylabel('\fontsize{14}p [Pa]');
title('\fontsize{14}Navrzeny tepelny obeh motoru (p-v diagram)')

hold on;

```

```

r_v = 287.1;
r_pl = 287.4;

% Urceni merneho objemu ze stavove rovnice
% -----
v_0c = r_v * T_0c / p_0c;
v_1c = r_v * T_1c / p_1c;
v_2c = r_v * T_2c / p_2c;
v_3c = r_pl * T_3c / p_3c;
v_4c = r_pl * T_4c / p_4c;
v_5c = r_pl * T_5c / p_5c;
v_6c = r_pl * T_6c / p_6c;

% Vykresleni jednotlivych stavu 0-6
% -----
plot([v_0c,v_1c,v_2c,v_3c,v_4c,v_5c,v_6c],[p_0c,p_1c,p_2c,p_3c,p_4c,p_5c,p_6c],...
    'LineStyle','none','Marker','square','MarkerSize',8,'MarkerEdgeColor',[0 0 0],...
    'MarkerFaceColor',[1 1 0]);

% Dej 0->1 (Ztrata celk. tlaku vzduchu na vstupu)
% -----
plot([v_0c,v_1c],[p_0c,p_1c],'r'); % usecka

% Dej 1->2 (Stlacení vzduchu v kompresoru)
% -----
n12 = log10(p_1c/p_2c) / log10(v_2c/v_1c); % urceni polytropické konst. n12
v12 = linspace(v_1c, v_2c, 100); % promenna - delit na 100 dilu
p12 = p_1c.*(v_1c./v12).^n12; % vypocet druhe souradnice
plot(v12,p12,'Color',[0 0 1],'LineWidth',3); % vykresleni polytropy

% Dej 2->3 (Ohrev plynu ve spalovací komore)
% -----
plot([v_2c,v_3c],[p_2c,p_3c],'Color',[0 0 1],'LineWidth',3); % usecka

% Dej 3->4 (Expanze plynu na turbine do pomyslného rezu 4)
% -----
n34 = log10(p_3c/p_4c) / log10(v_4c/v_3c); % urceni polytropické konst. n34
v34 = linspace(v_3c, v_4c, 100); % promenna - delit na 100 dilu
p34 = p_3c.*(v_3c./v34).^n34; % vypocet druhe souradnice
plot(v34,p34,'Color',[0 0 1],'LineWidth',3); % vykresleni polytropy

% Dej 4->5 (Expanze plynu na turbine od pomyslného rezu 4)
% -----
n45 = log10(p_4c/p_5c) / log10(v_5c/v_4c); % urceni polytropické konst. n45
v45 = linspace(v_4c, v_5c, 100); % promenna - delit na 100 dilu
p45 = p_4c.*(v_4c./v45).^n45; % vypocet druhe souradnice
plot(v45,p45,'Color',[0 0 1],'LineWidth',3); % vykresleni polytropy

% Dej 5->6 (Expanze plynu na vystupu z motoru)
% -----
plot([v_5c,v_6c],[p_5c,p_6c],'Color',[0 0 1],'LineWidth',3); %usecka

% Dej 6->0 (otevreny oběh)
% -----
plot([v_6c,v_0c],[p_6c,p_0c],'Color',[0 0 1],'LineStyle','--','LineWidth',2); %usecka

```

```
% NAVRH ZAKLADNICH ROZMERU ODSREDIVEHO KOMPRESORU
% Bc. Martin Poledno, brezen 2009
% -----

clear; clc;

% Zakladni udaje o kompresoru:
% -----
Qm = 1.1;           % [kg/s]      hmotnostni prtok vzduchu kompresorem
pi_kc = 4.1;        % [-]        stlaceni kompresoru
eta_kc = 0.8;       % [-]        izoentropicka ucinnost kompresoru

% Fyzikalni vlastnosti vzduchu:
% -----
kappa = 1.4;        % [-]        Poissonova konstanta
c_pv = 1005;        % [J/(kg*K)] izobaricka mer. tepelna kapacita
r = 287.1;          % [J/(kg*K)] merna plynova konstanta

% Odhadovane a volene hodnoty:
% -----
sigma_vst = 0.985;  % [-]        soucinitel zachovani celk. tlaku ve vstupni soustave
mi = 0.86;          % [-]        soucinitel skluzu (obvykle 0.86 - 0.92)
alfa = 0.05;        % [-]        soucinitel treci prace disku (obvykle 0,03 - 0,05)
c2r_ku_u2 = 0.28;   % [-]        pomer c2r/u2 (obvykle 0.25 - 0.35)
ni = 0.333;         % [-]        pomer D_1i/D_1e (obvykle 0.3 - 0.6)
ni2 = 0.6089;       % [-]        pomer D_1e/D_2 (obvykle 0.45 - 0.65)
ni3 = 1.1379;       % [-]        pomer D_3/D_2 (obvykle 1.05-1.15)
ni4 = 1.2626;       % [-]        pomer D_4/D_3 (obvykle 1.25-1.35)
xi_1 = 0.16;        % [-]        souc. dilci treci prace OK (obvykle 0.1 - 0.2)
xi_2 = 0.16;        % [-]        souc. dilci treci prace OK (obvykle 0.1 - 0.2)
xi_tr = 0.012;      % [-]        souc. treci prace BLD (obvykle 0.0075 - 0.015)
xi_dif = 0.20;      % [-]        souc. treci prace LD (voli se podle M_3c a theta)
t = 0.0013;         % [m]        sirka lopatky obezneho kola (obvykle 1.2mm - 3mm)
t_LD = 0.003;       % [m]        sirka lopatky LD (obvykle 2.5mm - 5mm)
delta = 0.0003;     % [m]        vule mezi lopatkami OK a skrini (0.3mm - 0.6mm)
ro3_ro2 = 1.09;     % [-]        pomer ro_3/ro_2 (obvykle 1.07 - 1.09)
z_LD = 24;          % [-]        pocet lopatek LD
c_4 = 106;          % [m/s]       vystupni rychlost z LD
c_5 = 100;          % [m/s]       vystupni rychlost z celeho kompresoru
n4 = 1.9;           % [-]        polytropicky exponent stlaceni na vystupu z komp.

% Pro nadmorskou vysku H=0 plati:
p_0 = 101325;       % [Pa]        standartni atmosfericky tlak
T_0 = 288;          % [K]        standartni atmosfericka teplota
ro_0 = p_0/(r*T_0);

M_0 = 0;            % [-]        rychlost letu (Machovo cislo)

disp('=====');
disp('                                OBEZNE KOLO                                ');
disp('=====');

% Celkovy tlak a teplota na vstupu do motoru
p_0c = p_0 * (1 + (kappa-1)/2 * M_0^2)^(kappa/(kappa-1));
```



```

T_0c = T_0 * (1 + (kappa-1)/2 * M_0^2);

% Celkovy tlak a teplota na vstupu do kompresoru
p_1c = p_0c * sigma_vst;
T_1c = T_0c;

% Vypocet adiabatice prace kompresoru
W_adc = c_pv * T_1c * (pi_kc^((kappa-1)/kappa)-1);

% Stanoveni obvodove rychlosti na vnejsim prumeru kola
u_2 = sqrt(W_adc/(eta_kc*(alfa+mi)));
c_2r = c2r_ku_u2 * u_2;
c_1a = c_2r;
c_1 = c_1a; % (vzduch vstupuje do kompresoru bez rozvireni)

% Veliciny na vstupu do kola na strednim prumeru
c_1s = c_1;
T_1 = T_1c - (c_1s^2)/(2*c_pv);
p_1 = p_1c * (T_1/T_1c)^(kappa/(kappa-1));
ro_1 = p_1/(r*T_1);

% Vnejsi prumer na vstupu do kola
D_1e = sqrt(4*Qm/(pi*c_1a*ro_1*(1-ni^2)));
D_1i = D_1e * ni;
D_1s = (D_1e + D_1i)/2;

% Prumer obehneho kola na vystupu
D_2 = D_1e / ni^2;

% Otacky kompresoru
n = 60 * u_2 / (pi * D_2);

% Rychlostni trojuhelnik
u_1s = n/60 * pi * D_1s;
u_1e = n/60 * pi * D_1e;
w_1s = sqrt(c_1^2 + u_1s^2);
w_1e = sqrt(c_1^2 + u_1e^2);

% Staticka teplota na vnejsim prumeru vstupu do kola
T_1e = T_0c - c_1^2/(2*c_pv);

% Machovo cislo relativniho proudu na vnejsim prumeru vstupu do kola
M_wle = w_1e / (20.05 * sqrt(T_1e)); % M_wle < 1
disp('Machovo cislo na vstupu do kola M_wle');
disp(M_wle);
if M_wle < 1
    disp('Ok. Na vstupu do kola je podzvukova rychlost proudu.');
```

```

else
    disp('POZOR: NA VSTUPU DO OBEZNEHO KOLA JE NADZVUKOVA RYCHLOST PROUDU!');
```

```

end
disp('-----');
```

```

% Uhel beta_1s na strednim vstupu
beta_1s = atand(c_1a/u_1s); % beta_1s > 25°
disp('Uhel beta_1s');
disp(beta_1s);
if beta_1s > 25
    disp('Ok. Uhel beta_1s > 25°.');
else
    disp('POZOR: UHEL BETA_1s < 25°!!!');
```

```

end
disp('-----');

% Volba poctu lopatek obezneho kola
if D_2 > 0.3
    z_k = 10 + 25*D_2;
else
    z_k = 10 + 30*D_2;
end
z_k = round(z_k);

% Pocet lopatek by mel byt sudy
if mod(z_k,2) == 1 % pokud je pocet lopatek lichy
    z_k = z_k + 1; % pricti 1
end
disp('Zvolen pocet lopatek OK z_k');
disp(z_k);
disp('-----');

% Kontrola voleneho soucinitele skluzu mi
mi_vyp = 1 / (1 + 2*pi/(3*z_k) / (1 - (D_1s/D_2)^2));
delta_mi = (mi_vyp - mi) / mi * 100;
disp('Rozdil mezi navrzenym a vypoctenym mi [%]');
disp(delta_mi);
if abs(delta_mi) > 2
    disp('CHYBA: NESOUHLASI NAVRZENE A VYPOCTNE MI!');
else
    disp('Ok. Navrzene a vypoctene mi souhlasí s presnosti na 2%.');
end
disp('-----');

% Stanoveni celkove teploty za kolem (zanedbani tepla odvedeneho do atmosfery)
T_2c = T_1c + W_adc/(eta_kc * c_pv);

% Urceni absolutni rychlosti na vystupu z kola
c_2u = mi * u_2;
c_2 = sqrt(c_2u^2 + c_2r^2);

% Stanoveni staticke teploty a Machova cisla na vystupu z kola
T_2 = T_2c - c_2^2/(2*c_pv);
M_c2 = c_2 / (20.05 * sqrt(T_2));
disp('Machovo cislo na vystupu z kola M_c2');
disp(M_c2);
disp('-----');

% Celkove ztraty pri prutoku vzduchu kolem
W_r1 = xi_1 * w_1s^2 / 2;
W_r2 = xi_2 * c_2r^2 / 2;
W_rd = alfa * u_2^2;
W_r = W_r1 + W_r2 + W_rd;

% Stanoveni polytropickeho exponentu stlaceni vzduchu
polytrop = kappa/(kappa-1) - W_r/(r*(T_2-T_1));

% Stanoveni statickeho tlaku a hustoty vzduchu na vystupu z kola
p_2 = p_1 * (T_2/T_1)^polytrop;
p_2c = p_2 * (T_2c/T_2)^(kappa/(kappa-1));
ro_2 = p_2 / (r*T_2);

% Stanoveni sirky kola na vystupu

```

```

tau_1 = 1 - z_k * t / (pi * D_2);
b_2 = Qm / (c_2r * ro_2 * pi * D_2 * tau_1);

disp('=====');
disp('                      BEZLOPATKOVY DIFUZOR                      ');
disp('=====');

% Urceni vstupni rychlosti do BLD
% Doporucuje se zvetsit sirku BLD oproti sirce kola o 0.6mm-0.8mm
b_2BLD = b_2 + 0.0007;
c_2uBLD = c_2u;
c_2rBLD = c_2r * tau_1 * b_2/b_2BLD;

% Stredni uhel na vstupu do BLD
alfa_2BLD = atand(c_2rBLD/c_2uBLD);
c_2BLD = c_2rBLD/sind(alfa_2BLD);

% Delka BLD
D_3 = D_2 * ni3;
b_3 = b_2BLD;

% Stredni uhel na vystupu
alfa_3 = atand(c_2rBLD/c_2uBLD + xi_tr/b_3*(D_3/2-D_2/2));

% Rychlost na vystupu z BLD
c_3r = c_2rBLD * D_2/D_3 * ro3_ro2;
c_3 = c_3r/sind(alfa_3);

% Staticka teplota vzduchu na vystupu z BLD
T_3c = T_2c;
T_3 = T_3c - c_3^2/(2*c_pv);

% Urceni treci prace v BLD
alfa_str = (alfa_3 + alfa_2BLD)/2;
W_rBLD = xi_tr/b_3*(c_2BLD^2 - c_3^2) / sind(alfa_str) * (D_3/2-D_2/2);

% Vypocet polytropickeho exponentu stlaceni
polytrop_BLD = kappa/(kappa-1) - W_rBLD/(r*(T_3-T_2));
n2 = polytrop_BLD / (polytrop_BLD - 1);

% Hustota vzduchu na vystupu z BLD
ro_3 = ro_2 * (T_3/T_2)^(1/(n2-1));
c_3rNOVE = c_2rBLD * D_2/D_3 * ro_3/ro_2;
disp('Rozdil mezi navrzenou a vypoctenou c_3r [%]');
disp((c_3rNOVE/c_3r-1)*100);
if abs((c_3rNOVE/c_3r-1)*100) > 1
    disp('CHYBA: NESOUHLASI RYCHLOST C_3r!');
else
    disp('Ok. C_3r souhlasi s presnosti na 1%.');
end
disp('-----');

% Tlak na vystupu z BLD
p_3 = p_2*(T_3/T_2)^polytrop_BLD;
p_3c = p_3*(T_3c/T_3)^(kappa/(kappa-1));

% Kontrola Machova cisla na vystupu z BLD
M_c3 = c_3 / (20.05 * sqrt(T_3));
disp('Machovo cislo na vystupu z BLD M_c3');

```

```

disp(M_c3);

disp('=====');
disp('                                LOPATKOVY DIFUZOR                                ');
disp('=====');

b_4 = b_3;
alfa_3k = alfa_3 + 0; % +/- 1

% Vystupni uhel proudu z LD
alfa_4 = round(alfa_3 + 15); % +(12-20)

% Uhel lopatky na vystupu
alfa_4k = alfa_4 + 3; % +(2-4)

% Urceni D_4
D_4 = D_3 * ni4;

% Kontrola pomeru prutocnych ploch A4/A3. Melo by byt cca 2.5
disp('Pomer prutocnych ploch A4/A3 by mel byt cca 2.5');
disp((D_4 * sind(alfa_4k)) / (D_3 * sind(alfa_3k)));
disp('-----');

% Delka stredni krivky lopatky
l = (D_4^2 - D_3^2) / 2 / sqrt(D_4^2 + D_3^2 - 2*D_3*D_4*cosd(alfa_3k + alfa_4k));

% Urcime uhel rozevreni kanalu LD
d_4 = sqrt(4*D_4*b_4*sind(alfa_4k)/z_LD);
d_3 = sqrt(4*D_3*b_3*sind(alfa_3k)/z_LD);
theta = 2*atand((d_4 - d_3)/(2*l));

% Podle theta a M_c3 volime soucintel ztrat xi_dif

% Volime vystupni rychlost z LD c_4

% Staticka teplota na vystupu z LD
T_4c = T_2c;
T_4 = T_4c - c_4^2/(2*c_pv);

% Treci prace v LD
W_rLD = xi_dif * (c_3^2 - c_4^2)/2;

% Polytropicky koeficient stlaceni
polytrop_LD = kappa/(kappa-1) - W_rLD/(r*(T_4-T_3));
n3 = polytrop_LD / (polytrop_LD - 1);

% Hustota vzduchu na vystupu z LD
ro_4 = ro_3 * (T_4/T_3)^(1/(n3-1));

% Kontrola vystupni rychlosti c_4
tau_2 = 1 - z_LD * t_LD / pi / D_4;
c_4NOVE = Qm / (ro_4 * sind(alfa_4k) * pi * D_4 * b_4 * tau_2);
disp('Rozdil mezi navrzenou a vypotenou c_4 [%]');
disp((c_4NOVE/c_4-1)*100);
if abs((c_4NOVE/c_4-1)*100) > 1
    disp('CHYBA: NESOUHLASI RYCHLOST C_4!!!');
else
    disp('Ok. C_4 souhlasi s presnosti na 1%.');
end

```

```

disp('-----');

% Tlak na vystupu z LD
p_4 = p_3 * (T_4/T_3)^polytrop_LD;
p_4c = p_4 * (T_4c/T_4)^(kappa/(kappa-1));

% Machovo cislo na vystupu z LD
M_c4 = c_4 / (20.05 * sqrt(T_4));
disp('Machovo cislo na vystupu z LD M_c4');
disp(M_c4);

disp('=====');
disp('                      VYSTUP Z KOMPRESORU                      ');
disp('=====');

% Stredni prumer na vystupu D_5
D_5 = D_4 + 2 * b_4;

% Volime vystupni rychlost c_5

% Stavove veliciny na vystupu
T_5c = T_4c;
T_5 = T_2c - c_5^2/(2*c_pv);
ro_5 = ro_4 * (T_5/T_4)^(1/(n4-1));
p_5 = p_4 * (T_5/T_4)^(n4/(n4-1));

% Urcime prutokovou plochu A_5
A_5 = Qm / (ro_5 * c_5);

% Urcime rozmery vystupu
D_51 = D_5 - A_5/(pi*D_5);
D_52 = 2*D_5 - D_51;

% Stanovime celkovy tlak na vystupu
p_5c = p_5 * (T_2c/T_5)^(kappa/(kappa-1));

disp('=====');
disp('                      KONTROLA STLACENI A UCINNOSTI                      ');
disp('=====');

% Kontrola stlaceni kompresoru
pi_kcVYP = p_5c/p_1c;
disp('Nove vypoctene stlaceni kompresoru');
disp(pi_kcVYP);
disp('Rozdil mezi navrzenym a vypoctenym stlacenim [%]');
disp((pi_kcVYP-pi_kc)/pi_kcVYP*100);
disp('-----');

% Kontrola ucinnosti kompresoru
eta_kcVYP = (pi_kcVYP^((kappa-1)/kappa)-1)/(T_2c/T_1c - 1);
disp('Nove vypoctena ucinnost kompresoru');
disp(eta_kcVYP);
disp('Rozdil mezi navrzenou a vypoctenou ucinnosti [%]');
disp((eta_kcVYP-eta_kc)/eta_kc*100);

% -----
% Vykresleni grafu prubehu termodynamickych velicin
%-----

```

```

% Příprava dat
p_c = [p_0c, p_1c, p_2c, p_3c, p_4c, p_5c];
p = [p_0, p_1, p_2, p_3, p_4, p_5];
T_c = [T_0c, T_1c, T_2c, T_3c, T_4c, T_5c];
T = [T_0, T_1, T_2, T_3, T_4, T_5];
ro = [ro_0, ro_1, ro_2, ro_3, ro_4, ro_5];

% Vykreslení grafu
figure1 = figure('Color',[1 1 1],...
    'Name','Průběh tlaků, teplot a měrné hmotnosti vzduchu v kompresoru');

AX1 = axes('FontSize',14,...
    'XGrid','on',...
    'XTick',[0 1 2 3 4 5],...
    'XTickLabel',{'0','1','2','3','4','5'},...
    'YColor',[0 0 0.6275],...
    'Parent',figure1);

hold on;
plot(0:5,p_c,'LineStyle','--','Color',[0.2 0.2 0.7],'LineWidth',3);
plot(0:5,p,'LineStyle','--','Color',[0.4 0.4 1],'LineWidth',3);
plot(0:5,ro.*10^5,'LineStyle','--','Color',[0.2 0.8 0.2],'LineWidth',3);
ylabel(['p_c, p [Pa]',char(10),' \rho . 10^5 [kg.m^{-3}]']);
legend('p_c','p','\rho','Location','NorthWest');
xlim([0 5]);
set(AX1,'ActivePositionProperty','position')

AX3 = axes('Position',get(AX1,'Position'),...
    'YAxisLocation','right',...
    'FontSize',14,...
    'XTick',[0 1 2 3 4 5],...
    'XTickLabel',{'0','1','2','3','4','5'},...
    'XGrid','on',...
    'YColor',[0.502 0 0],...
    'Color','none',...
    'Parent',figure1);

hold on;
plot(0:5,T_c,'LineStyle','--','Color',[0.7 0 0],'LineWidth',3);
plot(0:5,T,'LineStyle','--','Color',[1 0.3 0.3],'LineWidth',3);
xlim([0 5]);
ylim([280 900]);
ylabel('T_c, T [K]');
set(legend('T_c','T'),'Color','w')

```

```
% VYPOCET CHARAKTERISTIKY KOMPRESORU
% Bc. Martin Poledno, červenec 2009
% -----
clc; clear;
nactidiagramy;

% Zadani
% -----
PI_VYP = 4.1;
QM_VYP = 1.1;
ETA_VYP = 0.8;
kappa = 1.4;

% Zadani otacek, pro ktere pocitat charakteristiku
% -----
n = [0.7 0.76 0.82 0.88 0.94 1 1.05];

% Pocet bodu, ktere budou tvorit vetev charakteristiky
% -----
pocet_bodu = 20;

% Stanoveni pumpovni hranice pro otacky n = 1
% -----
c_pump = cti(DG8,1);
W_pump_opt = cti(DG3,c_pump);
eta_pump_max = ctiDG1(c_pump,1);

PI_PUMP_N1 = (1 + W_pump_opt * eta_pump_max * (PI_VYP)^((kappa-1)/kappa)-1)...
    ^(kappa/(kappa-1));
ETA_PUMP_N1 = ETA_VYP * eta_pump_max;
QM_PUMP_N1 = c_pump * QM_VYP * (PI_PUMP_N1/PI_VYP)^(1/3);

clear c_pump W_pump_opt eta_pump_max;

% Stanoveni pumpovni hranice pro vsechny vetve charakteristiky
% -----
for i = 1:length(n)
    eta_pump_n1 = cti(DG4a,n(i));
    KOMP_CHAR(i).rel_otacky = n(i);
    KOMP_CHAR(i).stlaceni(1) = (1 + eta_pump_n1 * n(i)^2 * ((PI_PUMP_N1)^...
        ((kappa-1)/kappa) - 1))^(kappa/(kappa-1));
    KOMP_CHAR(i).prutok(1) = QM_PUMP_N1 * sqrt((KOMP_CHAR(i).stlaceni(1) - 1.4)...
        /(PI_PUMP_N1 - 1.4));
    KOMP_CHAR(i).ucinnost(1) = ETA_PUMP_N1 * eta_pump_n1;
end

% Vypocet bodu vetve charakteristiky pro volene otacky
% -----

for i = 1:length(n)

    j = 0;
    eta_max_vyp = cti(DG4,n(i));
    c_pump = cti(DG8,n(i));
    W_pump_opt = cti(DG3,c_pump);
```

```

eta_pump_max = ctiDG1(c_pump,n(i));
c_max = interp1([0.7, 1.05],[1.3, 1.3],n(i),'linear'); % oriznuti max. c

krok = linspace(c_pump,c_max,pocet_bodu);

for c = krok;

    j = j + 1;
    W_opt = cti(DG3,c);
    eta_max = ctiDG1(c,n(i));

    KOMP_CHAR(i).stlaceni(j) = (1 + ((KOMP_CHAR(i).stlaceni(1))^(kappa-1)/kappa)-1)...
        / (W_pump_opt * eta_pump_max) * W_opt * eta_max)^(kappa/(kappa-1));

    KOMP_CHAR(i).prutok(j) = KOMP_CHAR(i).prutok(1) / c_pump * c...
        * (KOMP_CHAR(i).stlaceni(j)/KOMP_CHAR(i).stlaceni(1))^(1/3);

    KOMP_CHAR(i).ucinnost(j) = ETA_VYP * eta_max_vyp * eta_max;
end
end

% Vykresleni zavislosti stlaceni na prutoku
% -----

figure('Color',[1 1 1]);
axes('LineWidth',1,'FontSize',14);

xlabel('\bf\fontsize{16}Q_m [kg.s^{-1}]');
ylabel('\bf\fontsize{20}\pi\fontsize{16}_{kc} [-]');
title('\bf\fontsize{16}Charakteristika kompresoru \fontsize{20}\pi\fontsize{16}_{kc} = f(Q_m, n)');

hold on;
for i = 1:length(n)
    plot(KOMP_CHAR(i).prutok(1),KOMP_CHAR(i).stlaceni(1),'LineStyle','none',...
        'Marker','x','MarkerSize',14,'MarkerEdgeColor',[0.8 0.2 0]);
end

for i = 1:length(n)
    plot(KOMP_CHAR(i).prutok,KOMP_CHAR(i).stlaceni,'Color',[0.8 0.2 0],'LineWidth',2.5);
    text(KOMP_CHAR(i).prutok(end) + 0.00, KOMP_CHAR(i).stlaceni(end) + 0.1,...
        {'\fontsize{14}',num2str(n(i))});
end

% Vykresleni pumpovni cary
% -----

for i = 1:length(n)
    PC_X(i) = KOMP_CHAR(i).prutok(1);
    PC_Y(i) = KOMP_CHAR(i).stlaceni(1);
end

pc_x = linspace(PC_X(1), PC_X(end), 100);
pc_y = interp1(PC_X, PC_Y, pc_x,'pchip');

plot(pc_x, pc_y, '--','Color',[0.8 0.2 0],'LineWidth',1.0);

% Vyznaceni navrhoveho rezimu
% -----

```



```

plot(QM_VYP, PI_VYP,...
     'LineStyle','none','Marker','square','MarkerSize',8,'MarkerEdgeColor',[0 0 0],...
     'MarkerFaceColor',[1 1 0]);

% Vykresleni zavislosti ucinnosti na prutoku
% -----

figure('Color',[1 1 1]);
axes('LineWidth',1,'FontSize',14);
xlabel('\bf\fontsize{16}Q_m [kg.s^{-1}]');
ylabel('\bf\fontsize{20}\eta\fontsize{16}_{kc} [-]');
title('\bf\fontsize{16}Charakteristika kompresoru \fontsize{20}\eta\fontsize{16}_{kc} = f(Q_m, n)');
xlim('auto');
ylim([0.55 0.85]);
hold on;

for i = 1:length(n)
    plot(KOMP_CHAR(i).prutok,KOMP_CHAR(i).ucinnost,'Color',[0.75 0 0.75],'LineWidth',2.5);
    text(KOMP_CHAR(i).prutok(end) + 0.0, KOMP_CHAR(i).ucinnost(end) + 0,...
         {'\fontsize{14}' ,num2str(n(i))});
end

% Zapis promenne KOMP_CHAR do souboru 'kompresor.dat'
% -----
soubor = fopen('kompresor.dat', 'wt');
fprintf(soubor, '%i\n\n', length(n));
for i = 1:length(n)
    fprintf(soubor, '%3.2f\n', KOMP_CHAR(i).rel_otacky);
    fprintf(soubor, '%i\n', length(KOMP_CHAR(i).stlaceni));
    fprintf(soubor, '%i\n\n', length(KOMP_CHAR(i).ucinnost));
    for j = 1:length(KOMP_CHAR(i).stlaceni)
        fprintf(soubor, '%8.7f %8.7f\n', [KOMP_CHAR(i).prutok(j), KOMP_CHAR(i).stlaceni(j)]);
    end
    fprintf(soubor, '\n');
    for j = 1:length(KOMP_CHAR(i).ucinnost)
        fprintf(soubor, '%8.7f %8.7f\n', [KOMP_CHAR(i).prutok(j), KOMP_CHAR(i).ucinnost(j)]);
    end
    fprintf(soubor, '\n');
end
fclose(soubor);

```

```
% Nacte potrebné diagramy do pameti
% -----

global DG1;
soubor = fopen('Dg1.dat','r');

pocet_krivek = fscanf(soubor,'%i',1);
for i = 1:pocet_krivek
    DG1(i).rel_otacky = fscanf(soubor,'%f',1);
    pocet_bodu_krivky = fscanf(soubor,'%i',1);
    nactena_matice = fscanf(soubor,'%f %f', [2 pocet_bodu_krivky]);
    DG1(i).data = nactena_matice';
end

fclose(soubor);
clear pocet_krivek i pocet_bodu_krivky nactena_matice soubor;

global DG3;
soubor = fopen('Dg3.dat','r');
nactena_matice = fscanf(soubor,'%f %f', [2 Inf]);
DG3 = nactena_matice';
fclose(soubor);
clear nactena_matice soubor;

global DG4;
soubor = fopen('Dg4.dat','r');
nactena_matice = fscanf(soubor,'%f %f', [2 Inf]);
DG4 = nactena_matice';
fclose(soubor);
clear nactena_matice soubor;

global DG4a;
soubor = fopen('Dg4a.dat','r');
nactena_matice = fscanf(soubor,'%f %f', [2 Inf]);
DG4a = nactena_matice';
fclose(soubor);
clear nactena_matice soubor;

global DG8;
soubor = fopen('Dg8.dat','r');
nactena_matice = fscanf(soubor,'%f %f', [2 Inf]);
DG8 = nactena_matice';
fclose(soubor);
clear nactena_matice soubor;
```

Skript ke kapitole:	6.2 – CHARAKTERISTIKA ODSTŘEDIVÉHO KOMPRESORU
Název souboru:	<i>cti.m</i>

```
function [y] = cti(diagram,x);  
  
y = interp1(diagram(:,1), diagram(:,2), x, 'pchip');  
  
end
```

Skript ke kapitole:	6.2 – CHARAKTERISTIKA ODSTŘEDIVÉHO KOMPRESORU
Název souboru:	<i>ctiDG1.m</i>

```
function [y] = ctiDG1(x,n)  
global DG1;  
  
% KONTROLA ROZSAHU  
if (n < DG1(1).rel_otacky) || (n > DG1(end).rel_otacky)  
    y = NaN  
  
else  
    % V PORADKU  
    for i=1:length(DG1)-1  
        if (n >= DG1(i).rel_otacky) && (n <= DG1(i+1).rel_otacky)  
            nizsi = interp1(DG1(i).data(:,1),DG1(i).data(:,2), x, 'pchip');  
            vyssi = interp1(DG1(i+1).data(:,1),DG1(i+1).data(:,2), x, 'pchip');  
            y = (n - DG1(i).rel_otacky)/(DG1(i+1).rel_otacky - DG1(i).rel_otacky)...  
                *(vyssi-nizsi) + nizsi;  
        end  
    end  
end  
end
```

```

clear;
disp('-----');
disp('          V Y P O C E T   O S O V E   T U R B I N Y ');
disp('          1. s t u p e n ');
disp('-----');

W = 143711.33 / 0.8 + 99265;
x = 0.5337;
n = 48766;

Qm = 1.1;

r_pl = 287.4;
kappa = 1.33;
c_ppl = 1158;

% Celkový tlak a teplota na vstupu do turbíny
p_0c = 392833;
T_0c = 1000;

eta_m = 0.98; % mechanická účinnost, 0.97 až 0.98

mi = 1.6; % součinitel zatížení turbíny, 1.2 až 1.8
uls_k_c1 = 0.60; % poměr u_ls k c_1, 0.55 až 0.65
fi = 0.97; % součinitel ztráty rychlosti, 0.96 až 0.98
alfa_1 = 23; % úhel na středním průměru, 20 až 32 stupňů
ro_i = 0.09; % volíme 0 až 0.1
psi = 0.91; % součinitel ztráty rel. rychlosti, 0.88 až 0.94
h_ku_1 = 0.9; % poměr mezi sálkou a výskou lopatky, 0.8 až 0.9

% Celková práce turbíny
W_tc = W / eta_m;

% Práce stupně turbíny W_st1
W_st1 = W_tc * x;

% Práce na stupni, obvodová rychlost na středním průměru
u_ls = sqrt(W_st1/mi);

% Střední průměr na vstupu do rotoru
D_ls = u_ls * 60 / pi / n;

% Rychlost c_1
c_1 = u_ls / uls_k_c1;

% Součinitel rychlosti lambda1 na výstupu ze statoru
T_1c = T_0c;
a_kr1 = sqrt(2*kappa/(kappa+1)*r_pl*T_1c);
lambda1 = c_1 / a_kr1;

% Ideální rychlost c_1id
c_1id = c_1/fi;
lambda1_id = lambda1/fi;

```

```

% Stavove veliciny na vystupu ze statoru
p_l = p_0c * (1-(kappa-1)/(kappa+1)*lambda1_id^2)^(kappa/(kappa-1));
p_lc = p_l / ((1-(kappa-1)/(kappa+1)*lambda1^2)^(kappa/(kappa-1)));

T_l = T_lc - c_l^2 / (2*c_pp1);
ro_l = p_l / (r_pl * T_l);

c_las = c_l * sind(alfa_1);

% Plocha na vystupu ze statoru
A_l = Qm / (ro_l * c_las);
A_0 = A_l;

% Delka lopatky statoru
l_l = A_l / (pi * D_ls);

disp('l_l/D_ls:');
disp(l_l/D_ls);
if l_l/D_ls > 0.1 && l_l/D_ls < 0.34
    disp('OK. Pomer l_l/D_ls lezi v intervalu 0.1-0.34');
else
    disp('CHYBA! Pomer l_l/D_ls NELEZI v intervalu 0.1-0.34!!');
end
disp('-----');

% Prumery na vstupu do rotoru
D_li = D_ls - l_l;
D_le = D_ls + l_l;

% Relativni rychlost na vstupu do rotoru
w_l = c_l * sqrt(1 + (uls_k_c1)^2 - 2*uls_k_c1*cosd(alfa_1));

% Lopatkovani stupne s konstantnim vystupnim uhlem proudu ze statoru
c_lai = c_las * (D_ls/D_li)^(fi^2 * (cosd(alfa_1))^2);
c_lui = sqrt(c_l^2 - c_las^2) * (D_ls/D_li)^(fi^2 * (cosd(alfa_1))^2);
c_li = sqrt(c_lui^2 + c_lai^2);

% Relativni reakce
ro_rel = (c_li^2 - c_l^2) / c_li^2;

% Reakce na strednim prumeru
ro = ro_i + ro_rel * (1 - ro_i);

disp('ro:');
disp(ro);
if ro > 0.25 && ro < 0.35
    disp('OK. Soucinitel reakce RO lezi v intervalu 0.25-0.35');
else
    disp('CHYBA! Soucinitel reakce RO NELEZI v intervalu 0.25-0.35!!');
end
disp('-----');

% Adiabaticky spad na rotoru
h_02 = c_l^2 / (2 * fi^2) * ro / (1 - ro);

% Relativni rychlost na vystupu z rotoru
w_2 = psi * sqrt(w_l^2 + 2*h_02);

% Obvodova slozka relativni rychlosti

```

```

w_2u = W_st1/u_1s - (c_1*cosd(alfa_1) - u_1s);

% Slozky absolutni rychlosti na vystupu z rotoru
c_2u = w_2u - u_1s;
w_2a = sqrt(w_2^2 - w_2u^2);
c_2a = w_2a;

% Uhel absolutni rychlosti na vystupu z rotoru
alfa_2 = atand(c_2a/c_2u);

disp('alfa_2:');
disp(alfa_2);
if alfa_2 > 80 && alfa_2 < 100
    disp('OK. Uhel absolutni rychlosti lezi v intervalu 80° az 100°');
else
    disp('CHYBA!!! Uhel absolutni rychlosti NELEZI v intervalu 80° az 100°');
end
disp('-----');

% Stavove veliciny na vystupu z rotoru
c_2 = c_2a/sind(alfa_2);
T_2c = T_0c - W_st1/c_pp1;
T_2 = T_2c - c_2^2/(2*c_pp1);
p_2 = p_1*(1-h_02/(c_pp1*T_1))^(kappa/(kappa-1));
p_2c = p_2 * (T_2c/T_2)^(kappa/(kappa-1));
ro_2 = p_2 / (r_pl*T_2);

% Plocha na vystupu z rotoru
A_2 = Qm / (ro_2*c_2*sind(alfa_2));

% Tvar kanalu a delka lopatky
% Volime D_s = konst.
% D_2s = D_1s;
% l_2 = A_2/pi/D_2s
% D_2i = D_2s - l_2;
% D_2e = D_2s + l_2;

% Volime D_i = konst
% D_2i = D_1i;
% D_2e = sqrt(4*A_2/pi + D_2i^2);
% D_2s = (D_2i+D_2e)/2;
% l_2 = A_2/(pi*D_2s);

% Volime D_e = konst
D_2e = D_1e;
D_2i = sqrt(D_2e^2 - 4*A_2/pi);
D_2s = (D_2i+D_2e)/2;
l_2 = A_2/(pi*D_2s);

h_1 = h_ku_1*l_2;
gamma = atand((l_2 - l_1)/h_1);

disp('gamma:');
disp(gamma);
if gamma < 18
    disp('OK. Uhel gamma je mensi nez 18°');
else
    disp('CHYBA!!! Uhel gamma NENI mensi nez 18°');
end
disp('-----');

```

```

% Ucininnost stupne
disp('Ucininnost prvnioho stupne turbiny');
eta_st1 = (1 - T_2c/T_0c) / (1 - (p_2c/p_0c)^((kappa-1)/kappa));
disp(eta_st1);

% Tlakovy spad na prvnim stupni
disp('Tlakovy spad na prvnim stupni turbiny');
disp(p_0c/p_2c);

disp('-----');
disp('          V Y P O C E T    O S O V E    T U R B I N Y ');
disp('          2. s t u p e n ');
disp('-----');

mi = 1.6; % soucinitel zatizeni turbiny, 1.2 az 1.8
u3s_k_c3 = 0.58; % pomer u_3s k c_1, 0.55 az 0.65
fi = 0.971; % soucinitel ztraty rychlosti, 0.96 az 0.98
alfa_3 = 30; % uhel na strednim prumeru, 20 az 32 stupnu
ro_i = 0.05; % volime 0 az 0.1
psi = 0.915; % soucinitel rel. rychlosti, 0.88 az 0.94
h_ku_1 = 0.8; % pomer mezi sirkou a vyskou lopatky, 0.8 az 0.9

% Prace stupne turbiny W_st2
W_st2 = W_tc * (1 - x);

% Prace na stupen, obvodova rychlost na strednim prumeru
u_3s = sqrt(W_st2/mi);

% Stredni prumer na vstupu do rotoru
D_3s = u_3s * 60 / pi / n;

% Rychlost c_3
c_3 = u_3s / u3s_k_c3;

% Soucinitel rychlosti lambda1 na vystupu ze statoru
T_3c = T_2c;
a_kr3 = sqrt(2*kappa/(kappa+1)*r_pl*T_3c);
lambda3 = c_3 / a_kr3;

% Idealni rychlost c_3id
c_3id = c_3/fi;
lambda3_id = lambda3/fi;

% Stavove veliciny na vystupu ze statoru
p_3 = p_2c * (1-(kappa-1)/(kappa+1)*lambda3_id^2)^(kappa/(kappa-1));
p_3c = p_3 / ((1-(kappa-1)/(kappa+1)*lambda3^2)^(kappa/(kappa-1)));

T_3 = T_3c - c_3^2 / (2*c_pp1);
ro_3 = p_3 / (r_pl * T_3);

c_3as = c_3 * sind(alfa_3);

% Plocha na vystupu ze statoru
A_3 = Qm / (ro_3 * c_3as);
A_2 = A_3;

% Delka lopatky statoru
l_3 = A_3 / (pi * D_3s);

```

```

disp('l_3/D_3s:');
disp(l_3/D_3s);
if l_3/D_3s > 0.1 && l_3/D_3s < 0.34
    disp('OK. Pomer l_3/D_3s lezi v intervalu 0.1-0.34');
else
    disp('CHYBA! Pomer l_3/D_3s NELEZI v intervalu 0.1-0.34!!');
end
disp('-----');

% Prumery na vstupu do rotoru
D_3i = D_3s - l_3;
D_3e = D_3s + l_3;

% Relativni rychlost na vstupu do rotoru
w_3 = c_3 * sqrt(1 + (u_3s_k_c3)^2 - 2*u_3s_k_c3*cosd(alfa_3));

% Lopatkovani stupne s konstantnim vystupnim uhlem proudu ze statoru
c_3ai = c_3as * (D_3s/D_3i)^(fi^2 * (cosd(alfa_3))^2);
c_3ui = sqrt(c_3^2 - c_3as^2) * (D_3s/D_3i)^(fi^2 * (cosd(alfa_3))^2);
c_3i = sqrt(c_3ui^2 + c_3ai^2);

% Relativni reakce
ro_rel = (c_3i^2 - c_3^2) / c_3i^2;

% Reakce na strednim prumeru
ro = ro_i + ro_rel * (1 - ro_i);

disp('ro:');
disp(ro);
if ro > 0.25 && ro < 0.35
    disp('OK. Soucinitel reakce RO lezi v intervalu 0.25-0.35');
else
    disp('CHYBA! Soucinitel reakce RO NELEZI v intervalu 0.25-0.35!!');
end
disp('-----');

% Adiabaticky spad na rotoru
h_24 = c_3^2 / (2 * fi^2) * ro / (1 - ro);

% Relativni rychlost na vystupu z rotoru
w_4 = psi * sqrt(w_3^2 + 2*h_24);

% Obvodova slozka relatini rychlosti
w_4u = W_st2/u_3s - (c_3*cosd(alfa_3) - u_3s);

% Slozky absolutni rychlosti na vystupu z rotoru
c_4u = w_4u - u_3s;
w_4a = sqrt(w_4^2 - w_4u^2);
c_4a = w_4a;

% Uhel absolutni rychlosti na vystupu z rotoru
alfa_4 = atand(c_4a/c_4u);

disp('alfa_4:');
disp(alfa_4);
if alfa_4 > 80 && alfa_4 < 100
    disp('OK. Uhel absolutni rychlosti lezi v intervalu 80° az 100°');
else

```



```

disp('CHYBA!!! Uhel absolutni rychlosti NELEZI v intervalu 80° az 100°');
end
disp('-----');

% Stavove veliciny na vystupu z rotoru
c_4 = c_4a/sind(alfa_4);
T_4c = T_2c - W_st2/c_pp1;
T_4 = T_4c - c_4^2/(2*c_pp1);
p_4 = p_3*(1-h_24/(c_pp1*T_3))^(kappa/(kappa-1));
p_4c = p_4 * (T_4c/T_4)^(kappa/(kappa-1));
ro_4 = p_4 / (r_pl*T_4);

% Plocha na vystupu z rotoru
A_4 = Qm / (ro_4*c_4a);

% Tvar kanalu a delka lopatky
% Volime D_e = konst
D_4e = D_3e;
D_4i = sqrt(D_4e^2 - 4*A_4/pi);
D_4s = (D_4i+D_4e)/2;
l_4 = A_4/(pi*D_4s);

h_3 = h_ku_1*l_4;
gamma = atand((l_4 - l_3)/h_3);

disp('gamma:');
disp(gamma);
if gamma < 18
    disp('OK. Uhel gamma je mensi nez 18°');
else
    disp('CHYBA!!! Uhel gamma NENI mensi nez 18°');
end
disp('-----');

% Ucininnost stupne
disp('Ucininnost druheho stupne turbiny');
eta_st2 = (1 - T_4c/T_2c) / (1 - (p_4c/p_2c)^((kappa-1)/kappa));
disp(eta_st2);

% Tlakovy spad
disp('Tlakovy spad na druhem stupni turbiny');
disp(p_2c/p_4c);

disp('=====');

% Celkova ucinnost turbiny
disp('Celkova ucinnost turbiny');
disp((T_0c*(1-(p_2c/p_0c)^((kappa-1)/kappa))*eta_st1 + T_2c*(1-(p_4c/p_2c)...
    ^((kappa-1)/kappa))*eta_st2) / (T_0c*(1-(p_4c/p_0c)^((kappa-1)/kappa))));

%-----
% Vykresleni grafu prubehu termodynamickych velicin
%-----

% Priprava dat
T_0 = T_0c - c_1as^2 / (2*c_pp1);
p_0 = p_0c * (T_0/T_0c)^(kappa/(kappa-1));
ro_0 = p_0 / (r_pl * T_0);

```

```

p_c = [p_0c, p_1c, p_2c, p_3c, p_4c];
p = [p_0, p_1, p_2, p_3, p_4];
T_c = [T_0c, T_1c, T_2c, T_3c, T_4c];
T = [T_0, T_1, T_2, T_3, T_4];
ro = [ro_0, ro_1, ro_2, ro_3, ro_4];

% Vykresleni grafu
figure1 = figure('Color',[1 1 1],...
    'Name','Průběh tlaků, teplot a měrné hmotnosti vzduchu na turbine');

AX1 = axes('FontSize',14,...
    'XGrid','on',...
    'XTick',[0 1 2 3 4 5],...
    'XTickLabel',{'0','1','2','3','4','5'},...
    'YColor',[0 0 0.6275],...
    'Parent',figure1);

hold on;
plot(0:4,p_c,'LineStyle','--','Color',[0.2 0.2 0.7],'LineWidth',3);
plot(0:4,p,'LineStyle','--','Color',[0.4 0.4 1],'LineWidth',3);
plot(0:4,ro.*10^5,'LineStyle','--','Color',[0.2 0.8 0.2],'LineWidth',3);
ylabel(['p_c, p [Pa]',char(10),' \rho . 10^5 [kg.m^{-3}]']);
ylim([0 500e3]);
legend('p_c','p','\rho','Location','SouthWest');
set(AX1,'ActivePositionProperty','position')

AX3 = axes('Position',get(AX1,'Position'),...
    'YAxisLocation','right',...
    'FontSize',14,...
    'XTick',[0 1 2 3 4 5],...
    'XTickLabel',{'0','1','2','3','4','5'},...
    'XGrid','on',...
    'YColor',[0.502 0 0],...
    'Color','none',...
    'Parent',figure1);

hold on;
plot(0:4,T_c,'LineStyle','--','Color',[0.7 0 0],'LineWidth',3);
plot(0:4,T,'LineStyle','--','Color',[1 0.3 0.3],'LineWidth',3);
ylim([300 1050]);
ylabel('T_c, T [K]');
set(legend('T_c','T'),'Color','w')

```

Skript ke kapitole:	7.1 – NÁVRH ZÁKLADNÍCH ROMŽERŮ OSOVÉ TURBÍNY
Název souboru:	<i>rozdeleni.m</i>

```

% Pomocny skript pro nalezeni pomeru praci
% mezi prvni a druhy stupen turbiny
% -----

clear; clc;
i = 1;
promenna = 0.52: 0.002: 0.54;
for x = promenna
    x
    [p1(i),T1(i),D1e1(i),D1i1(i),D2i1(i)]=t1(x);
    [p2(i),T2(i),D1e2(i),D1i2(i),D2i2(i)]=t2((1-x),p1(i),T1(i));
    i = i + 1;
end
plot(promenna,D1e1,promenna,D1e2);
figure;
plot(promenna,D2i1,promenna,D1i2);

```

Skript ke kapitole:	7.2 – CHARAKTERISTIKA TURBÍNY
Název souboru:	<i>prepocet_turb.m</i>

```
% PREPOCET CHARAKTERISTIKY TURBINY
% Bc. Martin Poledno, duben 2010
% -----
clear; clc;

VYKRESLIT_VYSLEDEK = 1; % 0: prepocetna charakteristika se nevykresli
                        % 1: prepocetna charakteristika se vykresli

c_ppl = 1158;          % [J/(kg.K)]   izobaricka merna tep. kapacita plynu
T_3c = 1000;          % [K]   Celkova teplota pred turbinou
n_jmen = 48766;        % [min^-1] Jmenovite otacky motoru

% Nacte data ze souboru TURB.DAT do globalni promenne TURB_CHAR
% -----
global TURB_CHAR;
soubor = fopen('turb.dat','r');

pocet_krivek = fscanf(soubor,'%i',1);
for vetev = 1:pocet_krivek
    TURB_CHAR(vetev).otacky = fscanf(soubor,'%f',1);
    pocet_bodu = fscanf(soubor,'%i',1);
    nactena_matice = fscanf(soubor,'%f %f', [2 pocet_bodu]);
    TURB_CHAR(vetev).spad = nactena_matice(1,:);
    TURB_CHAR(vetev).prace = nactena_matice(2,:);
end

fclose(soubor);
clear pocet_krivek vetev pocet_bodu ...
    nactena_matice soubor;

if VYKRESLIT_VYSLEDEK ~= 0

% Vykresleni zavislosti merne prace turbiny na tlakovem spadu
% -----

figure('Color',[1 1 1]);
axes('LineWidth',1);

xlabel('\bf\fontsize{16}\pi\fontsize{12}_{tc} [-]');
ylabel('\bf\fontsize{12}W_{tc}/T_{3c} [J.kg^{-1}.K^{-1}]');
title('\bf\fontsize{12}Zavislost W_{tc} = f(\fontsize{16}\pi\fontsize{12}_{tc})');

hold on;
posun_y = 5;
for i = 1:length(TURB_CHAR)
    plot(TURB_CHAR(i).spad,TURB_CHAR(i).prace,'k.');
```

if i == length(TURB_CHAR) posun_y = 10; end

```
text(TURB_CHAR(i).spad(end) + 0.05, TURB_CHAR(i).prace(end) + posun_y,...
    {'\fontsize{10}', num2str(TURB_CHAR(i).otacky)});
end

for i=1:length(TURB_CHAR)
```

```

krok = linspace(min(TURB_CHAR(i).spad), max(TURB_CHAR(i).spad), 100);
interpolovano = interp1(TURB_CHAR(i).spad,TURB_CHAR(i).prace, krok, 'pchip');
plot(krok,interpolovano,'r');
end

end

% Prepecet zavislosti W_tc/T_3c=f(pi_tc) na rel_ochlazení=f(pi_tc)
% -----
for vetev = 1:length(TURB_CHAR)
    TURB_CHAR(vetev).otacky = TURB_CHAR(vetev).otacky * n_jmen / sqrt(T_3c);
    for j = 1:length(TURB_CHAR(vetev).spad)
        TURB_CHAR(vetev).rel_ochlazení(j) = TURB_CHAR(vetev).prace(j) * T_3c / ...
            (c_ppl * T_3c);
    end
end

% Provede zapis promenne TURB_CHAR do souboru 'TURBINA.DAT'%
% -----
soubor = fopen('turbina.dat','wt');
fprintf(soubor, '%i\n\n', length(TURB_CHAR));
for i = 1:length(TURB_CHAR)
    fprintf(soubor, '%4.3f\n', TURB_CHAR(i).otacky);
    fprintf(soubor, '%i\n', length(TURB_CHAR(i).spad));
    for j = 1:length(TURB_CHAR(i).spad)
        fprintf(soubor, '%8.7f %8.7f\n', [TURB_CHAR(i).spad(j), TURB_CHAR(i).rel_ochlazení(j)]);
    end
    fprintf(soubor, '\n');
end
fclose(soubor);

if VYKRESLIT_VYSLEDEK ~= 0

% Vykreslení závislosti relativního ochlazení na turbíně na tlakovém spadu
% -----
posun_y = 0.005;
figure('Color',[1 1 1]);
axes('LineWidth',1,'FontSize',14);

xlabel('\bf\fontsize{20}\pi\fontsize{16}_\{tc\} [-]');
ylabel('\bf\fontsize{16}\Delta T_\{tc\} / T_\{3c\} [-]');
title('\bf\fontsize{16}Charakteristika turbíny \Delta T_\{tc\} / T_\{3c\} = f(\fontsize{20}\pi\fontsize{16}_\{tc\}, n_\{par\})');

hold on;

for i = 1:length(TURB_CHAR)
    plot(TURB_CHAR(i).spad,TURB_CHAR(i).rel_ochlazení,'Color',[0.9 0.3 0.35],'LineWidth',2);
    if i == length(TURB_CHAR) posun_y = 0.015; end
    text(TURB_CHAR(i).spad(end) + 0.05, TURB_CHAR(i).rel_ochlazení(end) + posun_y,...
        {'\fontsize{14}',num2str(TURB_CHAR(i).otacky, '%4.0f')});
end

end
clear all;

```

```
% Vypocet charakteristiky trysky
% -----
clc; clear;

% -----
%          ZADANI
% -----
% Konstanty plynu
kappa = 1.33;
R = 287.4;

% Soucinitel kontrakce
mi = 0.92;

% Soucinitel ztraty rychlosti
fi = 0.98;

% Plocha prurezu na vystupu z trysky [m^2]
A_6 = 3.6200e-2;

% Celkova teplota a celkovy tlak na vystupu z turbiny
T_5c = 755.98;
p_5c = 102358;

% Definice vektoru pi_tr obsahujiciho hodnoty tlakovych spadu,
% pro ktere se bude pocitat charakteristika trysky
pi_tr = logspace(0,1,25)/200+1-0.005; % <1; 1.045> s logaritmicnym delenim

% -----
%          VYPOCET
% -----
% Vypocet konstanty K1
K1 = sqrt(kappa/R*(2/(kappa+1))^( (kappa+1)/(kappa-1) ));

% Vypocet funkce y
lambda_6id = sqrt((kappa+1)/(kappa-1)*(1-pi_tr.^((1-kappa)/kappa)));
lambda_6 = fi * lambda_6id;
y = ((kappa+1)/2)^(1/(kappa-1))*lambda_6./(1-(kappa-1)/(kappa+1)*lambda_6.^2);

% Vypocet parametru prutoku v zavislosti na pi_tr
par_Q = K1 * mi * A_6 * y ./ pi_tr;

% Skutecny hmotnostni prutok tryskou
Q = par_Q * p_5c ./ sqrt(T_5c);

% -----
%          VYKRESLENÍ GRAFU
% -----
% Vykresleni charakteristiky pi_tr = f(Q)
figure('Color',[1 1 1]);
axes('LineWidth',1);
xlabel('\bf\fontsize{12}Q_m [kg.s^{-1}]');
ylabel('\bf\fontsize{16}\pi\fontsize{12}_{tr} [-]');
title('\bf\fontsize{12}Charakteristika trysky \fontsize{16}\pi\fontsize{12}_{tr} = f(Q_m)');
```

```

hold on;
plot(Q, pi_tr, 'Color', [0.1 0.65 0.1], 'LineWidth', 2);

% Vykresleni charakteristiky pi_tr = f(par_Q)
figure('Color', [1 1 1]);
axes('LineWidth', 1);
xlabel('\bf\fontsize{12}Q_m*sqrt(T_{5c}) / p_{5c}');
ylabel('\bf\fontsize{16}\pi\fontsize{12}_{tr} [-]');
title('\bf\fontsize{12}Charakteristika trysky \fontsize{16}\pi\fontsize{12}_{tr} = f(Q_m*sqrt(T_{5c}) / p_{5c})');
hold on;
plot(par_Q, pi_tr, 'Color', [0.1 0.65 0.1], 'LineWidth', 2);

% -----
% ZAPIS DAT DO SOUBORU
% -----
% Zapis vypoctenych hodnot do souboru 'tryska.dat'
soubor = fopen('tryska.dat', 'wt');
for j = 1:length(pi_tr)
    fprintf(soubor, '%8.7f %8.7f\n', [par_Q(j), pi_tr(j)]);
end
fclose(soubor);

```

Skript ke kapitole: **9.1 – MEZINÁRODNÍ STANDARDNÍ ATMOSFÉRA**
Název souboru: ***MSA.m***

```

function [p,T] = MSA(H)
% Mezinarodni standardni atmosfera
% plati do vysky 11 000 m
% -----

%Definice konstant
p_H0 = 101325;    %[Pa]        normalni atmosfericky tlak vzduchu
g_H0 = 9.80665;   %[m/(s^2)]   normalni tihove zrychleni
r = 287.04;       %[J/(kg*K)]  plynova konstanta vzduchu
T_H0 = 288;       %[K]         normalni termodynamicka teplota vzduchu
lambda = -0.0065; %[K/m]       teplotni gradient

% Tlak vzduchu ve vysce H
p = p_H0.*(1 + lambda/T_H0.*H).^(-g_H0/(lambda*r));

% Teplota vzduchu ve vysce H
T = lambda.*H + T_H0;

```

```
% VYPOCET VYSKOVE CHARAKTERISTIKY TURBOHRIDELOVEHO MOTORU
% Bc. Martin Poledno, kveten 2010
% -----
clc; clear;

W_kc = 179639;
eta_kc = 0.8;
Qm0 = 1.1;
T3c0 = 1000;
p3c0 = 392833;
T6c0 = 755.98;
c60 = 65;
p0 = 101325;
eta_m = 0.98;

sigma_vst = 0.985;
sigma_sk = 0.96;
kappa = 1.4;
kappa_pl = 1.33;
c_pv = 1005;
c_ppl = 1158;

% Definice nadmorskych vysek H a rychlosti letu M
H = [0 1000 2000 3000 4000 5000];
M = [0 0.1 0.2 0.3 0.4 0.5];
% -----

for i=1:length(H)
    for j=1:length(M)

        % Odecteni p_H a T_H pro danou nadmorskou vysku H
        [pH, TH] = MSA(H(i));

        % Celkove parametry vzduchu pred motorem pro danou rychlost letu Ma
        pHc = pH * (1 + (kappa-1)/2 * M(j)^2)^(kappa/(kappa-1));
        THc = TH * (1 + (kappa-1)/2 * M(j)^2);

        % Celkove parametry vzduchu na vstupu do kompresoru
        plc = pHc * sigma_vst;
        T1c = THc;

        % Celkove parametry vzduchu za kompresorem
        p2c = plc * (1 + W_kc*eta_kc/(c_pv*T1c))^(kappa/(kappa-1));
        T2c = T1c + W_kc/c_pv;

        % Celkove parametry plynu pred turbinou
        p3c = p2c * sigma_sk;
        T3c = T3c0;

        % Hmotnostni prutok vzduchu motorem
        Qm = Qm0 * p3c / p3c0;

        % Vypocet staticke teploty plynu na vystupu z trysky pri H=0
        T60 = T6c0 - c60^2/(2*c_ppl);
```

```

% Vypocet ucinnosti expanze
eta_ex = (1 - T60/T3c0)/(1 - (pH/p3c)^((kappa_pl-1)/kappa_pl));

% Rychlost plynu na vytoku z vystupni soustavy
c6 = (p0/p3c0*(1 - eta_ex*(1 - (pH/p3c)^((kappa_pl-1)/kappa_pl)))...
      / (pH/p3c*(1 - eta_ex*(1 - (p0/p3c0)^((kappa_pl-1)/kappa_pl))))...
      * c60;

% Merna expanzni prace
W_ex = c_pp1 * T3c * (1 - (pH/p3c)^((kappa_pl-1)/kappa_pl)) * eta_ex;

% Uzitecny vykon na hrideli
P_ef(i,j) = Qm * (W_ex - W_kc/eta_m - c6^2/2);
end
end

% Vykresleni vyskove charakteristiky
% -----

figure('Color',[1 1 1]);
axes('LineWidth',1,'FontSize',14);

xlabel('\bf\fontsize{16}M [-]');
ylabel('\bf\fontsize{16}P_{ef} [ kW ]');
title('\bf\fontsize{16}Vyskova charakteristika motoru P_{ef} = f(M, H)');

xlim([0 M(end)+0.1]);
ylim([0 140]);

hold on;
for i=1:length(H)
    plot(M(:),P_ef(i,:)/1000,'Color',[0.85 0.1 0.0],'LineWidth',2);
    text(M(end) + 0.01, P_ef(i,end)/1000 + 4,{'\fontsize{14}' ,H(i)/1000});
end
for i=1:length(H)
    plot(0,P_ef(i,1)/1000,'LineStyle','none','Marker','o','MarkerSize',7,...
         'MarkerFaceColor',[0.85 0.8 0.0],'MarkerEdgeColor',[0.85 0.1 0]);
end

% Vykresleni zavislosti efektivni vykon vs. nadmorska vyska
% -----
figure('Color',[1 1 1]);
axes('LineWidth',1,'FontSize',14);
hold on;
xlabel('\bf\fontsize{16}H [km]');
ylabel('\bf\fontsize{16}P_{ef} [ kW ]');
title('\bf\fontsize{16}Zavislost P_{ef} = f(H)');

xlim([0 H(end)/1000+1]);
ylim([0 140]);
plot(H(:)/1000,P_ef(:,1)/1000,'Color',[0.85 0.1 0.0],'LineWidth',2);
plot(H(:)/1000,P_ef(:,1)/1000,'LineStyle','none','Marker','o','MarkerSize',7,...
     'MarkerFaceColor',[0.85 0.8 0.0],'MarkerEdgeColor',[0.85 0.1 0]);

```



```
% VYPOCET UZITECNEHO VYKONU P_ef V KONKRETNIM BODE KOMPRESOROVE
% CHARAKTERISTIKY ZADANEHO POMOCI (Qm, vetev)
% Bc. Martin Poledno, duben 2009
% -----

function [P_ef] = vypocetPef(Qm,vetev)
global KOMP_CHKA;
global TURB_CHKA;
global TRYSK_CHKA;

HRANICE_CHYBY = 0.5;

T3c_vyp = 1000; % [K] teplota plynu na vystupu ze spalovaci komory ve vypoctovem stavu
Qpl_vyp = 1.1; % [kg/s] prtok plynu ve vypoctovem stavu
pi_kc_vyp = 4.1; % [-]stlaci kompresoru ve vypoctovem stavu
n_vyp = 48766; % [1/min] vypoctove otacky motoru

sigma_vst = 0.985;
sigma_sk = 0.96;

r_v = 287.1; % plynova konstanta vzduchu
r_pl = 287.4; % plynova konstanta plynu
c_ppl = 1158; % izobaricka merna tepelna kapacita plynu
c_pv = 1005; % izobaricka merna tepelna kapacita vzduchu
kappa = 1.4; % kappa vzduchu
kappa_pl = 1.33; % kappa plynu
eta_mech = 0.98; % mechanicke ztraty na rotoru

H = 0;
Ma = 0;

%-----

[p0,T0] = MSA(H); % funkce atmosfera vrati tlak a teplotu ve vysce H [m] podle
                  % mezinarodni standartni atmosféry ISA

% Parametry vzduchu vstupu:

p0c = p0 * (1 + (kappa-1)/2 * Ma^2)^(kappa/(kappa-1));
T0c = T0 * (1 + (kappa-1)/2 * Ma^2);
naporove = p0c/p0;

plc = p0c * sigma_vst;
Tlc = T0c;

p3c_vyp = plc * pi_kc_vyp * sigma_sk;

%-----

% PARAMETR OTACEK KOMPRESORU
par_otacek_komp = KOMP_CHKA(vetev).otacky * n_vyp / sqrt(Tlc);

% PARAMETR PRUTOKU VZDUCHU KOMPRESOREM
[pi_kc, eta_kc] = cti_komp(Qm,vetev);
```

```

rel_ohrati_komp = (pi_kc^((kappa-1)/kappa)-1) / eta_kc;
par_prutoku_komp = Qm * sqrt(T1c) / plc;

% PARAMETR PRUTOKU PLYNU TURBINOU (predpokladame aerodynamicky ucpanou turbinu)
par_prutoku_turb = Qp1_vyp * sqrt(T3c_vyp) / p3c_vyp;

T3c_k_T1c = (par_prutoku_turb * sigma_sk * pi_kc / par_prutoku_komp)^2;

% PARAMETR OTACEK TURBINY
par_otacek_turb = par_otacek_komp * sqrt(1/T3c_k_T1c);

% Iteracni hledani rovnovahy mezi turbinou a tryskou
% -----

% Nalezene pocatecnich bodu A[a,fa], B[b,fb] pro iteracni metodu "regula falsi"
[a, b] = najdi_rozsah;

% bod A
rel_ochlazení_turb = cti_turb(a, par_otacek_turb);
par_prutoku_trysk = par_prutoku_turb * a * sqrt(1-rel_ochlazení_turb);
pi_tr = cti_trysk(par_prutoku_trysk);
pi_tr2 = 1/a * sigma_sk * pi_kc * sigma_vst * naporove;
fa = (pi_tr - pi_tr2)/pi_tr*100;

% bod B
fb = NaN;
while isnan(fb)
    b = b - 0.1;

    rel_ochlazení_turb = cti_turb(b, par_otacek_turb);
    par_prutoku_trysk = par_prutoku_turb * b * sqrt(1-rel_ochlazení_turb);
    pi_tr = cti_trysk(par_prutoku_trysk);
    pi_tr2 = 1/b * sigma_sk * pi_kc * sigma_vst * naporove;
    fb = (pi_tr - pi_tr2)/pi_tr*100;

end

%disp('Iteracni hledani rovnovahy mezi turbinou a tryskou');
%disp('----- ');
%disp(' Pocatecni body A[a,fa], B[b,fb] ');
%disp(' ');
%disp(' pi_tc rel_chyba[%]');
%disp(' ----- ');
%disp([a,fa]);
%disp([b,fb]);

% Vlastni vypočet aproximace
k = 0;
rel_chyba = Inf;
while abs(rel_chyba) > HRANICE_CHYBY
    k = k + 1;
    pi_tc = b - (b-a)/(fb-fa)*fb;
    rel_ochlazení_turb = cti_turb(pi_tc, par_otacek_turb);
    par_prutoku_trysk = par_prutoku_turb * pi_tc * sqrt(1-rel_ochlazení_turb);
    pi_tr = cti_trysk(par_prutoku_trysk);
    pi_tr2 = 1/pi_tc * sigma_sk * pi_kc * sigma_vst * naporove;
    rel_chyba = (pi_tr - pi_tr2)/pi_tr*100;

```

```

vysledky(k,1) = k;
vysledky(k,2) = pi_tc;
vysledky(k,3) = rel_chyba;

if fa * rel_chyba ~= 0
    if fa * rel_chyba > 0
        a = pi_tc;
        fa = rel_chyba;
    else
        b = pi_tc;
        fb = rel_chyba;
    end
else break;
end
end

%disp('  iterace      pi_tc      rel_chyba[%]');
%disp('  ----- ');
%disp(vysledky);

clear a b fa fb k vysledky;

% Prebytek výkonu
%disp('Užitečný výkon P_ef ve zvoleném bodě pracovní charakteristiky motoru');
P_ef = (par_prutoku_turb * c_ppl * rel_ochlazení_turb * eta_mech * sqrt(T3c_k_T1c) * ...
        sigma_sk * pi_kc - par_prutoku_komp * c_pv * rel_ohrati_komp) * plc * sqrt(T1c);

```

Skript ke kapitole: **10 – STANOVENÍ UŽITEČNÉHO VÝKONU MOTORU V NEVÝP. REŽIMECH**
Název souboru: ***izocary.m***

```

% PROGRAM PRO VÝPOČET ČAR KONSTANTIHO PREBYTKU VÝKONU
% Bc. Martin Poledno, duben 2010
% -----

clc;
%clear;
nacti_komp;
nacti_turb;
nacti_tr;

% Definování P_ef, pro které vykreslovat čary
P_ef = -60e3:20e3:140e3;

% Pro každou hodnotu P_ef
for j=1:length(P_ef)
    disp('-----');
    % Pro každou otákovou větev charakteristiky
    for vetev=1:length(KOMP_CHKA)
        disp('Užitečný výkon [kW]');
        disp(P_ef(j)/1000);
        disp('Otáková větev:');
        disp(KOMP_CHKA(vetev).otacky);

        % Hledání průsečíku izocary výkonu s otákovou větví komp. ch-ky
        Qm = fzero(@(x) vypocetPef(x,vetev)-P_ef(j),(KOMP_CHKA(vetev).stlaceni(1,1)...

```

```

+KOMP_CHKA(vetev).stlaceni(end,1))/2);

% Pokud prusecik lezi mimo rozsah otackove vetve, ulozi se NaN
if Qm < KOMP_CHKA(vetev).stlaceni(1,1)-0.05 || ...
    Qm > KOMP_CHKA(vetev).stlaceni(end,1)
    matice_Pef(1,vetev,j) = NaN;
    matice_Pef(2,vetev,j) = NaN;
    disp('Qm lezi mimo vetev kompresorove ch-ky.');
```

```

else
    % Ulozeni souradnic [Qm, pi_kc] pruseciku
    matice_Pef(1,vetev,j) = Qm;
    matice_Pef(2,vetev,j) = cti_komp(Qm, vetev);
    disp('Qm:');
    disp(Qm);
end
disp('-----');
end
end

```

Skript ke kapitole:	10 – STANOVENÍ UŽITEČNÉHO VÝKONU MOTORU V NEVÝP. REŽIMECH
Název souboru:	<i>cti_komp.m</i>

```

function [pi_kc,eta_kc] = cti_komp(Qm, vetev);
% Funkce nacita globalni promennou KOMP_CHKA a vraci interpolovanou hodnotu stlaceni
% a rel. otepleni na kompresoru v zavislosti na zvolenem Qmred a otackove vetvi

global KOMP_CHKA;

pi_kc = interp1(KOMP_CHKA(vetev).stlaceni(:,1), KOMP_CHKA(vetev).stlaceni(:,2),...
    Qm, 'pchip');
eta_kc = interp1(KOMP_CHKA(vetev).ucinnost(:,1), KOMP_CHKA(vetev).ucinnost(:,2),...
    Qm, 'pchip');
```

Skript ke kapitole:	10 – STANOVENÍ UŽITEČNÉHO VÝKONU MOTORU V NEVÝP. REŽIMECH
Název souboru:	<i>cti_turb.m</i>

```

function [rel_ochlazení_turb] = cti_turb(pi_tc, par_otacek_turb);
% Funkce nacita globalni promennou TURB_CHKA a vraci nadvakrat interpolovanou hodnotu
% relativního ochlazení na turbine v zavislosti na zvolenem tlakovem spadu
% na turbine a parametru otacek turbiny

global TURB_CHKA;

% KONTROLA ROZSAHU
if par_otacek_turb < TURB_CHKA(1).par_otacek
    rel_ochlazení_turb = interp1(TURB_CHKA(1).data(:,1),TURB_CHKA(1).data(:,2),...
        pi_tc, 'pchip');
elseif par_otacek_turb > TURB_CHKA(end).par_otacek
    rel_ochlazení_turb = interp1(TURB_CHKA(end).data(:,1),TURB_CHKA(end).data(:,2),...
        pi_tc, 'pchip');
else

% V ROZSAHU
for i=1:length(TURB_CHKA)-1
    % najdeme hodnoty [nizsi,vyssi], mezi kterymi lezi par_otacek_turb
    if (par_otacek_turb >= TURB_CHKA(i).par_otacek) && ...
        (par_otacek_turb <= TURB_CHKA(i+1).par_otacek)

```

```

% rel. ochlazení pro nižší hodnotu parametru otacek
nizsi = interp1(TURB_CHKA(i).data(:,1),TURB_CHKA(i).data(:,2),...
    pi_tc, 'pchip');
% rel. ochlazení pro vyšší hodnotu parametru otacek
vyssi = interp1(TURB_CHKA(i+1).data(:,1),TURB_CHKA(i+1).data(:,2),...
    pi_tc, 'pchip');
% lineární interpolace mezi hodnotami nižší a vyšší, vrátí
% hodnotu pro nami požadovaný parametr otacek
rel_ochlazení_turb = (par_otacek_turb - TURB_CHKA(i).par_otacek)/...
    (TURB_CHKA(i+1).par_otacek - TURB_CHKA(i).par_otacek)...
    *(vyssi-nizsi) + nizsi;

end
end
end

```

Skript ke kapitole:	10 – STANOVENÍ UŽITEČNÉHO VÝKONU MOTORU V NEVÝP. REŽIMECH
Název souboru:	<i>cti_trysk.m</i>

```

function [pi_tr] = cti_trysk(par_prutoku_trysk);
% Funkce načítá globální proměnnou TRYSK_CHKA a vrací interpolovanou
% hodnotu tlakového spádu na trysce v závislosti na parametru prouku
% tryskou

global TRYSK_CHKA;

if par_prutoku_trysk > max(TRYSK_CHKA.data(:,1))
    % funkce vrátí NaN, pokud bude par_prutoku_trysk větší než mezní hodnota
    pi_tr = NaN;
else
    pi_tr = interp1(TRYSK_CHKA.data(:,1), TRYSK_CHKA.data(:,2), par_prutoku_trysk,...
        'pchip');
end

```

Skript ke kapitole:	10 – STANOVENÍ UŽITEČNÉHO VÝKONU MOTORU V NEVÝP. REŽIMECH
Název souboru:	<i>nacti_komp.m</i>

```

% Načte data ze souboru KOMPRESOR.DAT do globální proměnné KOMP_CHKA

global KOMP_CHKA;
soubor = fopen('kompresor.dat','r');

% první číslo v souboru udává počet křivek charakteristiky
pocet_krivek = fscanf(soubor,'%i',1);
% pro každou křivku načte:
for i = 1:pocet_krivek
    % první číslo v oddílu popisuje otáčky křivky ch-ky, zápis do struktury
    KOMP_CHKA(i).otacky = fscanf(soubor,'%f',1);
    % druhé číslo v oddílu udává počet bodů závislosti stlačení
    pocet_bodu_stlacení = fscanf(soubor,'%i',1);
    % třetí číslo v oddílu udává počet bodů závislosti účinnosti
    pocet_bodu_ucinnost = fscanf(soubor,'%i',1);
    % načtení matic se souřadnicemi bodů
    nactena_matice_stlacení = fscanf(soubor,'%f %f', [2 pocet_bodu_stlacení]);
    nactena_matice_ucinnost = fscanf(soubor,'%f %f', [2 pocet_bodu_ucinnost]);
    % pomocné matice se transponují a uloží do struktury

```

```

        KOMP_CHKA(i).stlaceni = nactena_matice_stlaceni';
        KOMP_CHKA(i).ucinnost = nactena_matice_ucinnost';
end

fclose(soubor);
clear pocet_krivek i pocet_bodu_stlaceni pocet_bodu_ucinnost...
    nactena_matice_stlaceni nactena_matice_ucinnost soubor;

```

Skript ke kapitole: **10 – STANOVENÍ UŽITEČNÉHO VÝKONU MOTORU V NEVÝP. REŽIMECH**
 Název souboru: ***nacti_turb.m***

```

% Nacte data ze souboru TURBINA.DAT do globalni promenne TURB_CHKA

global TURB_CHKA;
soubor = fopen('turbina.dat','r');

% prvni cislo v souboru udava pocet krivek charakteristiky
pocet_krivek = fscanf(soubor,'%i',1);
% pro kazdou krivku nacti:
for i = 1:pocet_krivek
    % prvni cislo v oddilu udava parametr otacek krivky ch-ky
    TURB_CHKA(i).par_otacek = fscanf(soubor,'%f',1);
    % druhe cislo v oddilu udava pocet bodu, ktere tvori krivku
    pocet_bodu = fscanf(soubor,'%i',1);
    % nacteni matice se souradnicemi bodu
    nactena_matice = fscanf(soubor,'%f %f', [2 pocet_bodu]);
    % pomocna matice se transponuje a ulozi do struktury
    TURB_CHKA(i).data = nactena_matice';
end

fclose(soubor);
clear pocet_krivek i pocet_bodu nactena_matice soubor;

```

Skript ke kapitole: **10 – STANOVENÍ UŽITEČNÉHO VÝKONU MOTORU V NEVÝP. REŽIMECH**
 Název souboru: ***nacti_tr.m***

```

% Nacte data ze souboru TRYSKA.DAT do globalni promenne TRYSK_CHKA

global TRYSK_CHKA;
soubor = fopen('tryska.dat','r');

% nacteni matice se souradnicemi bodu
nactena_matice = fscanf(soubor,'%f %f', [2 Inf]);
% pomocna matice se transponuje a ulozi do struktury
TRYSK_CHKA.data = nactena_matice';

fclose(soubor);
clear nactena_matice soubor;

```

Skript ke kapitole:	10 – STANOVENÍ UŽITEČNÉHO VÝKONU MOTORU V NEVÝP. REŽIMECH
Název souboru:	<i>najdi_rozsah.m</i>

```
function [min_pi_tc, max_pi_tc] = najdi_rozsah;
% Najde minimalni a maximalni pi_tc nactene charakteristiky
global TURB_CHKA;

min_pi_tc = Inf;
max_pi_tc = -Inf;
for k=1:length(TURB_CHKA)
    a = min(TURB_CHKA(k).data(:,1));
    b = max(TURB_CHKA(k).data(:,1));
    if a < min_pi_tc;
        min_pi_tc = a;
    end
    if b > max_pi_tc;
        max_pi_tc = b;
    end
end
```

Skript ke kapitole:	10 – STANOVENÍ UŽITEČNÉHO VÝKONU MOTORU V NEVÝP. REŽIMECH
Název souboru:	<i>vysledky.m</i>

```
% Vykresleni vysledku

% ----- VYKRESLENÍ ZAVISLOSTI STLACENÍ -----

figure('Color',[1 1 1]);
axes('LineWidth',1,'FontSize',14);

hold on;
for i=1:length(KOMP_CHKA)
    plot(KOMP_CHKA(i).stlaceni(:,1), KOMP_CHKA(i).stlaceni(:,2), 'k.', 'MarkerSize',14);

    text(KOMP_CHKA(i).stlaceni(end,1) - 0.0, KOMP_CHKA(i).stlaceni(end,2)...
        - 0.02,{'\fontsize{14}',KOMP_CHKA(i).otacky},'Color',[0.0 0.0 0.0]);
end
title('\fontsize{18}Cary konstantního P_{ef} [kW]');
xlabel('\fontsize{18}Q_m [kg.s^{-1}]');
ylabel('\fontsize{24}\pi_{kc} \fontsize{18} [-]');

% Vykresleni pumpovni cary
% -----

for i = 1:length(KOMP_CHKA)
    PC_X(i) = KOMP_CHKA(i).stlaceni(1,1);
    PC_Y(i) = KOMP_CHKA(i).stlaceni(1,2);
end

pc_x = linspace(PC_X(1), PC_X(end), 100);
pc_y = interp1(PC_X, PC_Y, pc_x, 'pchip');

plot(pc_x, pc_y, '--', 'Color',[0.8 0.2 0], 'LineWidth',1.0);

% Vykresli interpolovana data kompresorove charakteristiky
```

```

% -----
hold on;
for i=1:length(KOMP_CHKA)
    krok = linspace(min(KOMP_CHKA(i).stlaceni(:,1)), ...
        max(KOMP_CHKA(i).stlaceni(:,1)), 100);
    interpolovano = interp1(KOMP_CHKA(i).stlaceni(:,1),...
        KOMP_CHKA(i).stlaceni(:,2), krok, 'pchip');
    plot(krok,interpolovano,'LineWidth',2,'Color',[0.75 0.25 0.25]);
end

% Vykresli izocary uzitecneho vykonu
% -----
if exist('matice_Pef')

for j=1:length(matice_Pef(1,1,:))
    cara_dP_X = linspace(min(matice_Pef(1,:,j)),max(matice_Pef(1,:,j)),200);
    cara_dP_Y = interp1(matice_Pef(1,matice_Pef(1,:,j)>0,j), ...
        matice_Pef(2,matice_Pef(1,:,j)>0,j) ,cara_dP_X,'pchip');
    odstिन = 1 - j/length(matice_Pef(1,1,:));
    if P_ef(j) == 0
        barva = [0.5 0.8 0]; % zelena pro P_ef = 0
    else
        barva = [1-odstin/2 0.2 0.2+odstin/2];
    end
    plot(cara_dP_X,cara_dP_Y,'LineWidth',2,'Color',barva);
    text(cara_dP_X(end) + 0.01, cara_dP_Y(end) + 0.14,{'\fontsize{14}'...
        P_ef(j)/1000},'Color',barva );
end
end

```